Notice of Allowability	Application No. 10/643,741	Applicant(s) SCOTT ET AL.	
	Examiner Arpan P. Savla	Art Unit 2185	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *amendment filed 11/15/07*.

2. ☒ The allowed claim(s) is/are *1,5,6,9,11,19-21,24,25,27 and 28*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. ____.
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: ____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date ____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date ____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☒ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☒ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 11/15/07
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☐ Interview Summary (PTO-413), Paper No./Mail Date ____.
7. ☒ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other ____.

EXAMINER'S AMENDMENT

An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37.CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Thomas Brennan (Reg. No. 35,075) on February 4, 2008.

- Amended **claim 1** reads as follows:

1. An apparatus comprising:

a memory interface;

a plurality of queues connected to the memory interface, including a first queue and a second queue, wherein each of the plurality of queues holds pending memory requests and enforces an ordering in the commitment of the pending memory requests to memory;

one or more instruction-processing circuits, wherein each instruction-processing circuit is operatively coupled through the plurality of queues to the memory interface and wherein each of the plurality of instruction-processing circuits inserts one or more memory requests into at least one of the queues based on a first memory operation instruction, inserts a first synchronization marker into the first queue and inserts a second synchronization marker into the second queue based on a synchronization

operation instruction and inserts one or more memory requests into at least one of the queues based on a second memory operation instruction; and

a first synchronization circuit, operatively coupled to the first plurality of queues, that selectively halts processing of further memory requests from the first queue based on the first synchronization marker reaching a predetermined point in the first queue until the corresponding second synchronization marker reaches a predetermined point in the second queue;

wherein each of the memory requests is a memory reference, wherein the memory reference is generated as a result of instructions by the instruction-processing circuits,

wherein the first queue is used for only synchronization markers and vector memory references, and the second queue is used for only synchronization markers and scalar memory references,

wherein the synchronization operation instruction is an Lsync V,S-type instruction,

wherein the instruction-processing circuits include a data cache and wherein the Lsync V,S-type instruction prevents subsequent scalar references from accessing the data cache until all vector references have been sent to an external cache and all vector writes have caused any necessary invalidations of the data cache.

- **Claim 2** has been canceled

- **Claim 3** has been canceled

- **Claim 4** has been canceled

- **Claim 10** has been canceled

- Amended **claim 11** reads as follows:

11. A method comprising:

providing a memory interface;

providing a plurality of queues connected to the memory interface, including a first queue and a second queue, wherein each of first plurality of queues holds pending memory requests and enforces an ordering in the commitment of the pending memory requests to memory;

providing one or more instruction-processing circuits, wherein each instruction-processing circuit is operatively coupled through the plurality of queues to the memory interface;

inserting one or more memory requests into at least one of the queues based on a first memory operation instruction executed in one of the instruction-processing circuits;

inserting a first synchronization marker into the first queue and inserting a second synchronization marker into the second queue based on a synchronization operation instruction executed in one of the instruction-processing circuits;

inserting one or more memory requests into at least one of the queues based on a second memory operation instruction based on a second memory operation instruction executed in one of the instruction-processing circuits;

processing memory requests from the first queue; and

selectively halting further processing of memory requests from the first queue based on the first synchronization marker reaching a predetermined point in the first queue until the corresponding second synchronization marker reaches a predetermined point in the second queue;

wherein each of the memory requests is a memory reference,

wherein the first queue stores only synchronization markers and vector memory references, and wherein the second queue stores only synchronization markers and scalar memory references,

wherein the synchronization operation instruction is an Lsync V,S-type instruction,

wherein providing the instruction- processing circuits includes providing a data cache and wherein performing the Lsync V,S type instruction includes preventing subsequent scalar references from accessing the data cache until all vector references have been sent to an external cache and all vector writes have caused any necessary invalidations of the data cache.

- **Claim 12** has been canceled

- **Claim 13** has been canceled

- **Claim 14** has been canceled

- **Claim 20** has been canceled

- Amended **claim 21** reads as follows:

21. An apparatus comprising:

a memory interface;

a plurality of queues connected to the memory interface, including a first queue and a second queue, wherein each of the plurality of queues holds pending memory requests and enforces an ordering in the commitment of the pending memory requests to memory;

one or more instruction-processing circuits, wherein each instruction-processing circuit is operatively coupled through the plurality of queues to the memory interface and wherein each of the plurality of instruction-processing circuits includes:

means for inserting one or more memory requests into at least one of the queues based on a first memory operation instruction executed in one of the instruction-processing circuits;

means for inserting a first synchronization marker into the first queue and inserting a second synchronization marker into the second queue based on a synchronization operation instruction executed in one of the instruction-processing circuits;

means for inserting one or more memory requests into at least one of the queues based on a second memory operation instruction based on a second memory operation instruction executed in one of the instruction-processing circuits;

means for processing memory requests from the first queue;

and means for selectively halting further processing of memory requests from the first queue based on the first synchronization marker reaching a predetermined point in the first queue until the corresponding second synchronization marker reaches a predetermined point in the second queue;

wherein each of the memory requests is a memory reference,

wherein means for inserting to the first queue operates for only vector memory requests and synchronizations, and means for inserting to the second queue operates for only scalar memory requests and synchronizations,

wherein the synchronization operation instruction is an Lsync V,S-type instruction,

wherein the instruction-processing circuits include a data cache and wherein the Lsync V,S-type instruction prevents subsequent scalar references from accessing the data cache until all vector references have been sent to an external cache and all vector writes have caused any necessary invalidations of the data cache.

- **Claim 22** has been canceled

- **Claim 23** has been canceled

- Amended **claim 24** reads as follows:

24. A system comprising:

a plurality of processors, including a first processor and a second processor, wherein each of the processors includes:

a memory interface;

a plurality of Lsync queues connected to the memory interface, including a first Lsync queue and a second Lsync queue,

wherein each of the plurality of Lsync queues holds pending memory requests and enforces an ordering in the commitment of the pending memory requests to memory;

one or more instruction-processing circuits, wherein each instruction-processing circuit is operatively coupled through the plurality of Lsync queues to the memory

interface and wherein each of the plurality of instruction-processing circuits inserts one or more memory requests into at least one of the Lsync queues based on a first memory operation instruction, inserts a first Lsync synchronization marker into the first Lsync queue and inserts a second Lsync synchronization marker into the second Lsync queue based on a synchronization operation instruction, and inserts one or more memory requests into at least one of the Lsync queues based on a second memory operation instruction; and

a Lsync synchronization circuit, operatively coupled to the plurality of Lsync queues, that selectively halts processing of further memory requests from the first Lsync queue based on the first Lsync synchronization marker reaching a predetermined point in the first Lsync queue until the corresponding second Lsync synchronization marker reaches a predetermined point in the second Lsync queue; and

one or more Msync circuits, wherein each of the Msync circuits is connected to the plurality of processors and wherein each of the Msync circuits includes:

a plurality of Msync queues, including a first Msync queue and a second Msync queue, each of the plurality of Msync queues for holding a plurality of pending memory requests received from the Lsync queues, wherein the first Msync queue stores only Msync synchronization markers and memory requests from the first processor, and the second Msync queue stores only Msync synchronization markers and memory requests from the second processor; and

an Msync synchronization circuit, operatively coupled to the plurality of Msync queues, that selectively halts further processing of the memory requests from the first

Msync queue based on an Msync synchronization marker reaching a predetermined point in the first Msync queue until a corresponding Msync synchronization marker from the second processor reaches a predetermined point in the second Msync queue;

wherein each of the memory requests is a memory reference, wherein the memory reference is generated as a result of execution of instructions by instruction-processing circuits in each processor,

wherein each processor includes a data cache and wherein each Msync synchronization circuit includes an external cache, wherein the data cache and the external cache are used to perform an Lsync V,S type instruction, wherein the Lsync V,S type instruction prevents subsequent scalar references from accessing the data cache until all vector references have been sent to the external cache in a corresponding Msync synchronization circuit and all vector writes have caused any necessary invalidations of the data cache.

- **Claim 26** has been canceled

- Amended **claim 27** reads as follows:

27. A method comprising:

providing a plurality of processors, including a first processor and a second processor, wherein each of the processors includes a memory interface, a plurality of Lsync queues connected to the memory interface, including a first Lsync queue and a

second Lsync queue, wherein each of the plurality of Lsync queues holds pending memory requests and enforces an ordering in the commitment of the pending memory requests to memory, and one or more instruction-processing circuits, each of the instruction-processing circuits operatively coupled through the plurality of Lsync queues to the memory interface;

providing one or more Msync circuits, wherein each of the Msync circuits is connected to the plurality of processors and wherein each of the Msync circuits includes a plurality of Msync queues, including a first Msync queue and a second Msync queue, each of the plurality of Msync queues operatively coupled to the plurality of Lsync queues in one of the plurality of processors;

inserting one or more memory requests into at least one of the Lsync queues based on a first memory operation instruction executed in one of the instruction-processing circuits;

inserting a first Lsync synchronization marker into the first Lsync queue and inserting a second Lsync synchronization marker into the second Lsync queue based on a synchronization operation instruction executed in one of the instruction-processing circuits;

inserting one or more memory requests into at least one of the Lsync queues based on a

second memory operation instruction executed in one of the instruction-processing circuits; processing memory requests from the first Lsync queue; selectively halting further processing of memory requests from the first Lsync queue based on the

first Lsync synchronization marker reaching a predetermined point in the first Lsync queue until the corresponding second Lsync synchronization marker reaches a predetermined point in the second Lsync queue;

inserting Msync synchronization markers and memory requests received from the Lsync queues in the first processor into the first Msync queue;

inserting Msync synchronization markers and memory requests received from the Lsync queues in the second processor into the second Msync queue; and

selectively halting further processing of the memory requests from the first Msync queue based on an Msync synchronization marker reaching a predetermined point in the first Msync queue until a corresponding Msync synchronization marker from the second processor reaches a predetermined point in the second Msync queue;

wherein each of the memory requests is a memory reference,

wherein selectively halting further processing of memory requests from the first Lsync queue includes performing an Lsync V,S type instruction, wherein performing the Lsync V,S type instruction includes preventing subsequent scalar references from accessing a data cache in the processor until all vector references have been sent to an external cache in a corresponding Msync synchronization circuit and all vector writes have caused any necessary invalidations of the data cache.

- **Claim 29** has been canceled

Allowable Subject Matter

Claims 1, 5, 6, 9, 11, 19-21, 24, 25, 27, and 28 are allowed.

The following is an examiner's statement of reasons for allowance:

Upon further consideration, the combination of Smith/Chen/Cray as well as any of the prior art of record fails to disclose the combination including the limitations of:

(**Claim 1**) "...wherein the first queue is used for only synchronization markers and vector memory references, and the second queue is used for only synchronization markers and scalar memory references, wherein the synchronization operation instruction is an Lsync V,S-type instruction, wherein the instruction-processing circuits include a data cache and wherein the Lsync V,S-type instruction prevents subsequent scalar references from accessing the data cache until all vector references have been sent to an external cache and all vector writes have caused any necessary invalidations of the data cache."

(**Claim 11**) "...wherein the first queue stores only synchronization markers and vector memory references, and wherein the second queue stores only synchronization markers and scalar memory references, wherein the synchronization operation instruction is an Lsync V,S-type instruction, wherein providing the instruction-processing circuits includes providing a data cache and wherein performing the Lsync V,S type instruction includes preventing subsequent scalar references from accessing the data cache until all vector references have been sent to an external cache and all vector writes have caused any necessary invalidations of the data cache."

(**Claim 21**) "...wherein means for inserting to the first queue operates for only vector memory requests and synchronizations, and means for inserting to the second

queue operates for only scalar memory requests and synchronizations, wherein the synchronization operation instruction is an Lsync V,S-type instruction, wherein the instruction-processing circuits include a data cache and wherein the Lsync V,S-type instruction prevents subsequent scalar references from accessing the data cache until all vector references have been sent to an external cache and all vector writes have caused any necessary invalidations of the data cache."

(**Claim 24**) "...wherein each processor includes a data cache and wherein each Msync synchronization circuit includes an external cache, wherein the data cache and the external cache are used to perform an Lsync V,S type instruction, wherein the Lsync V,S type instruction prevents subsequent scalar references from accessing the data cache until all vector references have been sent to the external cache in a corresponding Msync synchronization circuit and all vector writes have caused any necessary invalidations of the data cache."

(**Claim 27**) "...wherein selectively halting further processing of memory requests from the first Lsync queue includes performing an Lsync V,S type instruction, wherein performing the Lsync V,S type instruction includes preventing subsequent scalar references from accessing a data cache in the processor until all vector references have been sent to an external cache in a corresponding Msync synchronization circuit and all vector writes have caused any necessary invalidations of the data cache."

As dependent claims **5, 6, 9, 19, 20, 25, and 28** depend from an allowable base claim, they are at least allowable for the same reasons as noted above.

Any comments considered necessary by Applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Arpan P. Savla whose telephone number is (571) 272-1077. The examiner can normally be reached on M-F 8:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Sanjiv Shah can be reached on (571) 272-4098. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Arpan Savla
Art Unit 2185
February 5, 2008

SANJIV SHAH
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100